Paper No. 26

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte WEI MING HU
and JUAN R. LOAIZA

Appeal No. 2004-0268
Application 09/223,660[1]

ON BRIEF

Before BARRETT, BLANKENSHIP, and SAADAT, Administrative Patent Judges.

BARRETT, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal under 35 U.S.C. § 134(a) from the final rejection of claims 1-72.

We reverse.

[1] Application for patent filed December 30, 1998, entitled "Method and System for Diagnostic Preservation of the State of a Computer System."

BACKGROUND

The invention relates to a method and system for diagnostic preservation of the state of a computer system after a failure.

Claim 1 is reproduced below. Note that the "s," underlined below, was inadvertently left off the term "resources" in the amendment mailed January 16, 2002 (Paper No. 8).

> 1. A method of diagnosing a computer system after a failure comprising:
>
> preserving in place the state of a first set of system resources after the failure occurs in the computer system;
>
> accessing the computer system by utilizing a second set of system resources; and
>
> diagnosing the failure by analyzing one or more resources from the first set of system resources.

THE REFERENCES

The examiner relies on the following references:

Randell et al. (Randell)	4,164,017	August 7, 1979
Tandon	5,485,573	January 16, 1996
Chung et al. (Chung)	6,195,760	February 27, 2001 (filed July 20, 1998)

THE REJECTION

Claims 1-72 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Tandon, Chung, and Randell.

We refer to the final rejection (Paper No. 13) and the examiner's answer (Paper No. 20) (pages referred to as "EA__") for a statement of the examiner's rejection, and to the appeal brief (Paper No. 19) (pages referred to as "Br__") and reply

brief (Paper No. 23) (pages referred to as "RBr__") for a statement of appellants' arguments thereagainst.

OPINION

Claims 1-72 are grouped to stand or fall together with claim 1 being the representative claim (Br3). We agree with this grouping. The limitation at issue is "preserving in place the state of a first set of system resources after the failure occurs in the computer system." Independent claims 1, 21, and 41 contain this limitation. Independent claims 12, 22, 52 recite "preserving in place the state of one or more resources on the first computer system" after detecting a failure on the first computer system, which is substantially identical.

The examiner first combines Tandon and Chung. The examiner finds and concludes (EA4):

> Tandon and Chung do not expressly teach preserving in place the state of a first set of system resources after the failure occurs in the computer system. However, Rendell [sic] teaches a method for recovery from a failure with a block of a computer program (as admitted by the applicant from the paper 8). Further, Randell teaches preserving the state of each item of information after [the] error occurs (see column 1 lines 47-50 of Randell). It would have been obvious to include such teaching preserving the state of a computer system after the failure occurs into the system of Tandon and Chung to diagnose and analyze the reason for failure and thereby prevent[] errors in [the] future.

Appellants note that claim 1 recites "preserving <u>in place</u> the state of a first set of system resources after the failure occurs in the computer system" (emphasis added). It is argued that Randell teaches storing information on the original state of

a system and then restoring the system to an original state using the stored information when a failure occurs and, thus, does not preserve the state "in place" (Br4).

The examiner interprets, for the first time in the answer, the limitation, "preserving in place the state of a first set of system resources after the failure occurs in the computer system," as "a primary computer having resources such as network and I/O resources is running and when there is a failure, the previous state of the primary computer would then save into a particular location of memory (in place) after a failure" (EA6-7). The examiner states that appellants have never defined what the phrase "in place" means and has not provided a special definition in the specification (EA7).

Appellants reply that this is the first time that the examiner has asserted this claim interpretation (RBr2). It is agreed that claims must be interpreted as broadly as possible, but must be consistent with the specification. It is argued that the specification specifically discusses that copying the state of the primary computer to a memory location is inadequate, problematic, and costly, and is not done by the present invention (RBr3). Appellants note that the examiner interprets the limitation to mean storing "the previous state," but the limitation makes it clear that it is the state of the system resources after the failure occurred, not the state before the

failure occurred (RBr3). It is noted that the dictionary defines "in place" as "in an original position ... in the same spot without forward or backward movement" (RBr3). It is argued that Randell only teaches restoring the apparatus to its previous state after failure occurs and does not teach "preserving in place the state" after the failure occurs (RBr5-6).

We agree with appellants that "in place" has a well known common meaning. As defined in the dictionary provided by appellants, "in place" means "in an original position ... in the same spot without forward or backward movement," e.g., a band marching in place. The specification is consistent with this definition and describes that "[t]he state of the system resources are frozen" (page 7, lines 3-4); "freezing the state of an application or system upon error handling" (page 7, lines 7-8); "preserving or 'freezing' the state of this set of resources" (page 8, lines 14-15). The fact that "in place" is not defined in the specification does not mean that it does not have its common meaning. There is a "heavy presumption in favor of the ordinary meaning of claim language." Johnson Worldwide Assocs. v. Zebco Corp., 175 F.3d 985, 989, 50 USPQ2d 1607, 1610 (Fed.Cir.1999). It is only where an applicant decides to be his own lexicographer and give a term a definition that differs from the conventional definition, that it must be clearly set forth in the specification. See Beachcombers v. WildeWood Creative Prod.,

Inc., 31 F.3d 1154, 1158, 31 USPQ2d 1653, 1656 (Fed. Cir. 1994). The examiner has not provided any support for giving "in place" a broader interpretation. Therefore, "preserving in place the state of a first set of system resources after the failure occurs in the computer system" must be interpreted to mean that the state of the first set of system resources is maintained in its original position, i.e., frozen.

Tandon relates to detection and identification of errors in programs which execute on multiple host processors and more particularly to the collection of the necessary data for determining the source of errors in a multi-host data base management system (DBMS) (col. 1, lines 21-25). Tandon discloses that after detecting an error the contents of memory allocated to the DBMS is saved to a dump file, which is then available for analysis by a system programmer for isolating the source of the error (col. 5, lines 30-33). Saving memory contents to a dump file is not "preserving in place."

Chung relates to fault protection using multiple backup copies that are maintained in a fail-over state (abstract). Upon failure of an application module, the last stored state of that failed application module is retrieved from the memory of a checkpoint server and provided to a new primary application module for continued processing (col. 4, lines 44-48). While appellants describe a fail-over procedure for a DBMS, appellants

preserve the state of resources on the primary system, while Chung does not. Chung does not teach "preserving in place."

Thus, we agree with the examiner that neither Tandon nor Chung discloses "preserving in place."

Randell discloses that a program can be considered to consist of "blocks" of identifiable operations (col. 1, lines 9-15). On completion of a block, it is frequently necessary to restore the apparatus to the state it was in at the beginning of the block (col. 1, lines 15-17). Randell discloses an apparatus for use with a computer process comprising a set of program blocks, the apparatus being operative to restore the "state" of a plurality of "items of information" at the beginning of each block where the items of information change state during each block so that the conditions prevailing at the beginning of the block can be restored (col. 1, lines 23-31). The "items of information" may be variables and the "states" of the items of information can be the values assigned to the variable (col. 1, lines 52-57). The program is constructed from a number of identifiable operations for error recovery (col. 2, lines 24-28). A "recovery block" is a set of operations grouped together for purposes of permitting recovery from errors (col. 2, lines 29-31). Each recovery block contains a "primary block" and an "acceptance test" and may contain one or more "alternate blocks," where each primary or alternate block may itself contain

a recovery block (col. 2, lines 25-36). This structure is shown in Fig. 1. An "acceptance test" is a section of program which is invoked upon exit from a primary or alternate block in a recovery block and provides a binary decision as to whether the operation required of the recovery block as been performed acceptably (col. 2, lines 50-58). If the primary of a recovery block is rejected, the program must perform an alternate, and if the alternate fails, the entire recovery block fails and the system is restored to the state it was in before the block started (col. 2, lines 50-68; col. 3, lines 3-27). The apparatus has a first memory for storing the most recent states of the items of information, and means for transferring the existing state of an item of information from the first memory to a second memory when the state of the item of information changes and restoring the state of an item of information to the first memory when the acceptance test fails so that the state prevailing at the beginning of the block can be restored (col. 1, lines 26-44). Thus, the state of the variables is "rolled back."

Randell teaches "preserving the states" of items of information (abstract) by "restoring the state" of a set of program variables after a failure occurs, but does not teach "preserving <u>in place</u> the state of a first set of system resources after the failure occurs in the computer system." The state is not frozen "in place" in the computer, but is moved between the

first and second memories. We agree with appellants that Randell does not store "in place" but "necessitates the use and movement of resources, which could change or modify the state of the resources that are suppose[d] to be preserved" (RBr3). Also, we agree with appellants that the limitation "preserving in place the state of a first set of system resources after the failure occurs in the computer system" "makes it clear that the state of the system resources that is preserved is the state the system resources were in after the failure occurred and is not the state the system resources were in before the failure occurred, as asserted by the Examiner" (RBr3), and as disclosed by Randell. Accordingly, we conclude that the examiner has failed to establish a prima facie case of obviousness. The rejection of claims 1-72 is reversed.

REVERSED

LEE E. BARRETT
Administrative Patent Judge)

HOWARD B. BLANKENSHIP
Administrative Patent Judge) BOARD OF PATENT APPEALS AND INTERFERENCES

MAHSHID D. SAADAT
Administrative Patent Judge)

BINGHAM, MCCUTCHEN LLP
THREE EMBARCADERO
18TH FLOOR
SAN FRANCISCO, CA 94111-4067